

FieldView では、FVX によるバッチ処理を行うことができますが、
ここでは、CFD 解析の計算ループ中に画像出力する事例をご案内いたします。
* 事例には OpenFOAM を利用しております。

HT032_sample.pw

Pointwise を利用した メッシュデータサンプルです。

以下のディレクトリ内には、すでに polyMesh データは含まれています。

HT032/

OpenFOAM の計算事例ディレクトリ(ケースディレクトリ)です。

constant/	メッシュなどのディレクトリ(今回は関係なし)
orig0/	初期条件を入れたディレクトリ(今回は関係なし)
origMesh/	作業したメッシュデータを格納するディレクトリ(今回は関係なし)
system/	計算メソッド、計算条件類 ディレクトリ(今回は関係なし)
orgCreateImage.fv	サンプル FVX
transient_fv.sh	サンプル FVX を一定時刻置きに実行するシェルスクリプト

《使い方》 ケースディレクトリ内に移動し、以下のコマンドを順次入力します

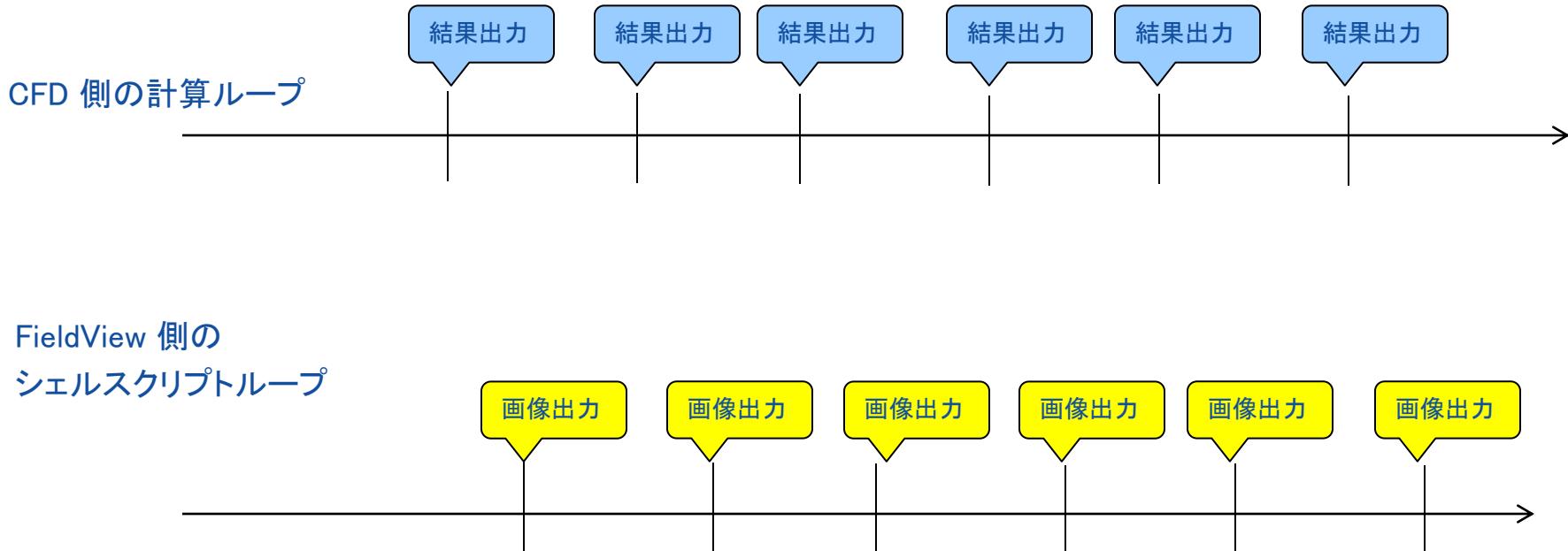
```
make clean      [enter]    … 前に流した計算データをクリアする  
make init      [enter]    … origMesh 内のメッシュデータを constant/polyMesh へコピー  
make fgrun      [enter]    … Makefile 内で指定されたソルバを実行  
* 本 サンプルでは buoyantBoussinesqPimpleFoam を利用(シングルコア)
```

計算が流れることを確認したら、
別のコンソールウインドウで、同じディレクトリへ移動し、

```
./transient.sh  [enter]
```

あるいは

```
source ./transient_fv.sh [enter]
```



今回のサンプルシェルスクリプトは CFD のジョブとは別個に稼働します。

上記のように、タイミングよくシェルスクリプトが稼働するように パラメータを設定する必要があります。

ソルバ側のデータ出力タイミングに関する箇所

system/ControlDict ファイル内

```

ファイル(E) 編集(E) 表示(V) 検索(S) ウィンドウ(W) マクロ(M) その他(O)
0 100 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 1600 1700 1800 1900 2000 2100 2200 2300 2400 2500 2600 2700 2800 2900 3000 3100 3200 3300 3400 3500 3600 3700 3800 3900 4000 4100 4200 4300 4400 4500 4600 4700 4800 4900 5000 5100 5200 5300 5400 5500 5600 5700 5800 5900 6000 6100 6200 6300 6400 6500 6600 6700 6800 6900 7000 7100 7200 7300 7400 7500 7600 7700 7800 7900 8000 8100 8200 8300 8400 8500 8600 8700 8800 8900 9000 9100 9200 9300 9400 9500 9600 9700 9800 9900 10000 10100 10200 10300 10400 10500 10600 10700 10800 10900 11000 11100 11200 11300 11400 11500 11600 11700 11800 11900 12000 12100 12200 12300 12400 12500 12600 12700 12800 12900 13000 13100 13200 13300 13400 13500 13600 13700 13800 13900 14000 14100 14200 14300 14400 14500 14600 14700 14800 14900 15000 15100 15200 15300 15400 15500 15600 15700 15800 15900 16000 16100 16200 16300 16400 16500 16600 16700 16800 16900 17000 17100 17200 17300 17400 17500 17600 17700 17800 17900 18000 18100 18200 18300 18400 18500 18600 18700 18800 18900 19000 19100 19200 19300 19400 19500 19600 19700 19800 19900 20000 20100 20200 20300 20400 20500 20600 20700 20800 20900 21000 21100 21200 21300 21400 21500 21600 21700 21800 21900 22000 22100 22200 22300 22400 22500 22600 22700 22800 22900 23000 23100 23200 23300 23400 23500 23600 23700 23800 23900 24000 24100 24200 24300 24400 24500 24600 24700 24800 24900 25000 25100 25200 25300 25400 25500 25600 25700 25800 25900 26000 26100 26200 26300 26400 26500 26600 26700 26800 26900 27000 27100 27200 27300 27400 27500 27600 27700 27800 27900 28000 28100 28200 28300 28400 28500 28600 28700 28800 28900 29000 29100 29200 29300 29400 29500 29600 29700 29800 29900 29999

55
56 application buoyantBoussinesqPimpleFoam;
57
58 startFrom startTime;
59 //startFrom latestTime;
60
61 startTime 0;
62
63 stopAt endTime;
64
65 endTime 10;
66 //endTime 5000;
67
68 deltaT 0.025;
69 //deltaT 1.0;
70
71 writeControl timeStep;
72
73 //writeInterval 100;
74 writeInterval 100;
75
76 purgeWrite 0;
77
78 writeFormat ascii;
79
80 writePrecision 6;
81
82 writeCompression uncompressed;
83
84 timeFormat general;
85
86 timePrecision 6;
87
88 runTimeModifiable true;
89
90 adjustTimeStep yes;
91 //adjustTimeStep no;
92
93 maxCo 0.5;

```

結果出力の
イタレーションインターバル

Courant 数を自動調整する機能を on
時間刻みが変動型となります

シェルスクリプト側の画像出力タイミングに関する箇所 transient_fv.sh ファイル内

```

1 #!/bin/bash
2 ↓
3 fv_exec="/usr/local/FieldView/fv13/fv/bin/fv"↓
4 FileFVX="/orgCreateImage.fvx"↓
5 SleepTime="10"↓
6 ↓
7 ↓
8 #make fgrun↓
9 ↓
10 nCount=1↓
11 while true; do↓
12 ↓
13 nCount=$((nCount+1))↓
14 echo "nCount : "$nCount↓
15 ↓
16 ##### FV-UNS convert only latest time↓
17 foamToFieldview9 -latestTime↓
18 ↓
19 ##### detect latest FV-UNS file name↓
20 LatestUnsFile=`ls -la ./Fieldview/*_uns | head -n1 | awk '{printf($8, "%c")}'`↓
21 echo "Latest FV-UNS: "${LatestUnsFile}↓
22 ↓
23 sed "s@strFileName@${LatestUnsFile}@g" ${FileFVX} > ./tmp.fvx↓
24 sed "s@numF@${nCount}@g" ./tmp.fvx > ./CreateImage.fvx↓
25 ↓
26 ##### execute FVX batch↓
27 $fv_exec -batch -fvx ./CreateImage.fvx↓
28 ↓
29 echo `date` "done:"↓
30 sleep ${SleepTime}↓
31 ↓
32 done↓

```

HT032: CFD 計算 loop 中に FVX 処理を行う

FIELDVIEW

處理風景

9月 9日 (金) 16:53

```
yoshida@vmaeolia:~/OpenFOAM/yoshida-2.0.x/run/work/SupportPage/FieldView/HT032
ファイル(F) 編集(E) 表示(G) ブックマーク(B) ヘルプ(H)
HT032
yoshida@vmaeolia:~/OpenFOAM/yoshida-2.0.x/run/work/SupportPage/FieldView/HT032
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
time step continuity errors : sum local = 2.72441e-14, global = 5.20355e-16, cumulative = 1.2059e-08
DILUPBiCG: Solving for epsilon, Initial residual = 1.8646e-05, Final residual = 3.00886e-07, No Iterations 1
DILUPBiCG: Solving for k, Initial residual = 5.39771e-05, Final residual = 1.93479e-07, No Iterations 1
ExecutionTime = 21.96 s ClockTime = 22 s

Calculating averages
ゴミ
Time = 0.0144009

Courant Number mean: 0.000124411 max: 0.499691
deltaT = 0.000329947
DILUPBiCG: Solving for T, Initial residual = 2.61101e-05, Final residual = 1.05975e-05
DICPCG: Solving for p_rgh, Initial residual = 0.00597544, Final residual = 1.05975e-05
time step continuity errors : sum local = 1.53406e-10, global = -1.07923e-10
DICPCG: Solving for p_rgh, Initial residual = 0.00244669, Final residual = 1.05975e-05
time step continuity errors : sum local = 2.71942e-14, global = 4.99698e-16
DILUPBiCG: Solving for epsilon, Initial residual = 1.85604e-05, Final residual = 1.05975e-05
DILUPBiCG: Solving for k, Initial residual = 5.36753e-05, Final residual = 1.05975e-05
ExecutionTime = 22.49 s ClockTime = 23 s

Calculating averages
ゴミ
Time = 0.0147308

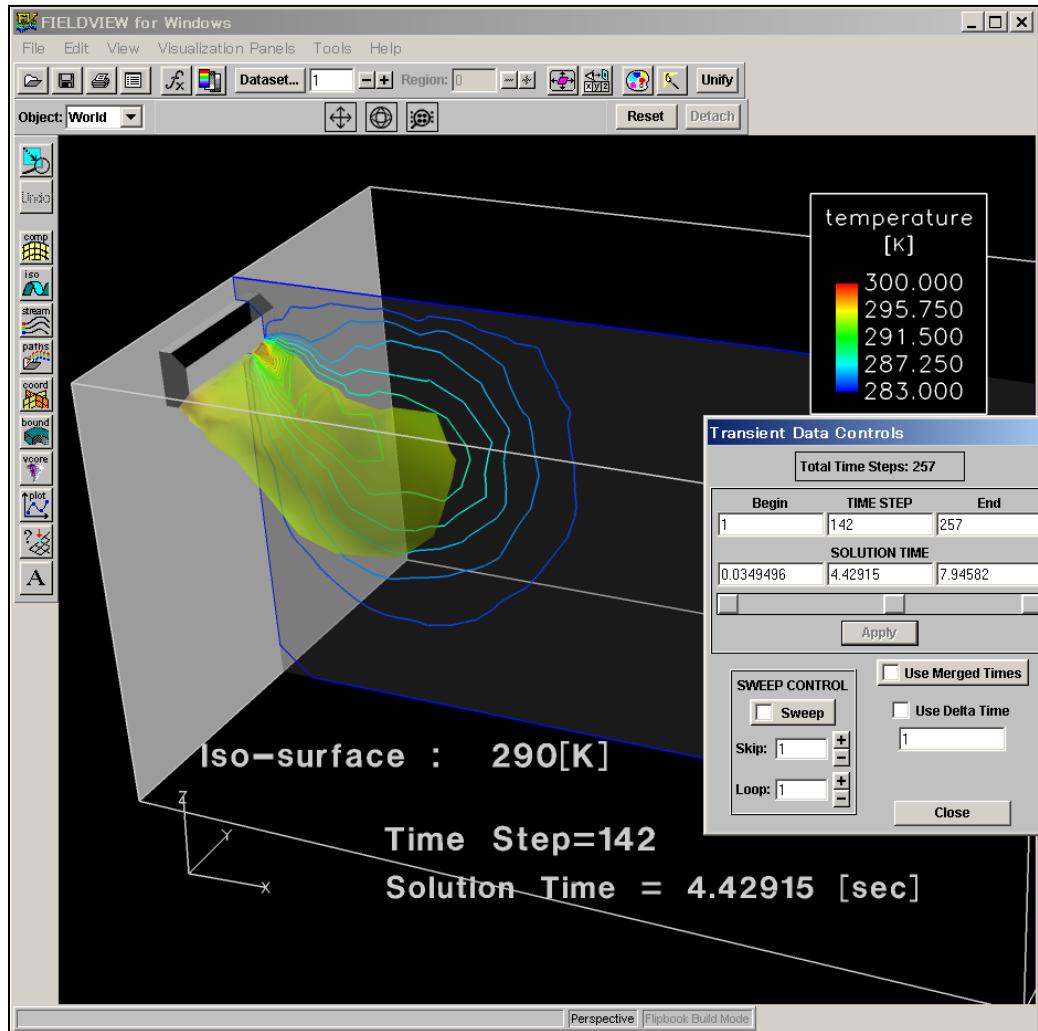
Courant Number mean: 0.00012467 max: 0.499676
deltaT = 0.000330161
DILUPBiCG: Solving for T, Initial residual = 2.5944e-05, Final residual = 1.05975e-05
DICPCG: Solving for p_rgh, Initial residual = 0.00581248, Final residual = 1.05975e-05
time step continuity errors : sum local = 1.51621e-10, global = -1.08452e-10
DICPCG: Solving for p_rgh, Initial residual = 0.00239949, Final residual = 1.05975e-05
time step continuity errors : sum local = 2.71521e-14, global = 4.7704e-16, cumulative = 1.20568e-08
DILUPBiCG: Solving for epsilon, Initial residual = 1.84661e-05, Final residual = 3.1008e-07, No Iterations 1
DILUPBiCG: Solving for k, Initial residual = 5.33865e-05, Final residual = 1.93479e-07, No Iterations 3
ExecutionTime = 23.05 s ClockTime = 23 s

Calculating averages
ゴミ
Time = 0.015061

Courant Number mean: 0.00012493 max: 0.499668
deltaT = 0.000330381
DILUPBiCG: Solving for T, Initial residual = 2.57878e-05, Final residual = 7.02971e-07, No Iterations 1
DICPCG: Solving for p_rgh, Initial residual = 0.005669, Final residual = 5.28627e-05, No Iterations 85
time step continuity errors : sum local = 1.44245e-10, global = -1.07054e-12, cumulative = 1.20558e-08
```

 VINAS
Visual Integration & Numerical Analysis Systems

解析事例: エアコンを模擬した 空調計算事例を使っています。



サンプルで
実際に出力し簡易画像

